import org.junit.jupiter.api.BeforeEach;

import org.junit.jupiter.api.Test;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;

import org.springframework.boot.test.context.SpringBootTest;

import org.springframework.http.MediaType;

import org.springframework.test.web.servlet.MockMvc;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.status;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.jsonPath;

@SpringBootTest

@AutoConfigureMockMvc

public class ProductRoutesTests {

@Autowired

private MockMvc mockMvc;

@Autowired

private ProductRepository productRepository;

private Product product1;

private Product product2;

@BeforeEach

public void setUp() {

// Create and save test products

product1 = new Product("Test Product A", 10.0, true);

product2 = new Product("Test Product B", 20.0, false);

productRepository.save(product1);

productRepository.save(product2);

}

@Test

public void testListProductsByName() throws Exception {

// Perform a GET request to retrieve products by name

mockMvc.perform(get("/products")

.param("name", "Test Product A") // Query parameter to filter by name

.contentType(MediaType.APPLICATION\_JSON))

.andExpect(status().isOk())

.andExpect(jsonPath("$.length()").value(1)) // Expect one product

.andExpect(jsonPath("$[0].name").value("Test Product A")); // Check name of the retrieved product

}

}